Attachment to BC-xxxx

Proceedings of the IASTED International Conference
Control'97 which was held May 28-31, 1997, Cancun, Mexico

FUZZY TECHNIQUES FOR CONTROLLING FLEXIBLE MANUFACTURING SYSTEMS

Paolo Dadone*, Hugh F. VanLandingham* and Bruno Maione**

*The Bradley Department of Electrical Engineering, Virginia Polytechnic Institute and State University, Blacksburg, Virginia 24061 - 0111 U.S.A.
**Dipartimento di Elettrotecnica ed Elettronica, Politecnico di Bari, Via E. Orabona 4, 70125 Bari, Italy

Keywords: Fuzzy sets, flexible manufacturing systems, evolutionary programming, multiple attribute decision making.

Abstract: The present paper describes an application of fuzzy logic systems (FLS) and fuzzy multiple attribute decision making (MADM) techniques for the scheduling of flexible manufacturing systems (FMS). A simulator-based reinforcement learning approach, that uses evolutionary programming techniques, is used as a design procedure for the fuzzy scheduler (FS). The resulting evolutionary fuzzy scheduler (EFS) is compared to commonly used heuristic rules and is found to perform better, especially in case of high arrival rates.

1. INTRODUCTION

An FMS is a production system consisting of identical multipurpose numerically controlled machines (workstations), automated material handling system, tools, load and unload stations, inspection stations, storage areas and a hierarchical control system ([5]). The short term scheduling (or control) of an FMS can be divided into four macro components: sequencing, timing, routing and priority setting [1]. In the design of a "controller" that takes care of the short term scheduling, several issues must be considered: the multiple objective nature of the problem [10], the large variability among plants, the need for adaptive control systems and the NP-hard nature of the scheduling problem. This makes typical operation research methods inadequate and a common way of realizing the controller is by using some heuristic rules ([10]).

A relatively new approach to the scheduling problem comes from the emerging field of "intelligent manufacturing". In this approach some intelligent control techniques are employed for the scheduling problem. Most of the approaches lack a systematic and general design procedure based on the multiple objective nature of the problem. All the attempts towards intelligent manufacturing show that some kind of "human reasoning" is necessary to achieve good scheduling. It is the authors' opinion that the importance of "common sense" and "human experts" in scheduling, together with fuzzy logic ability to mimic human reasoning, along with the ease of dealing with linguistic variables makes it a very suitable and powerful tool for scheduling in an FMS.

Among all the possible scheduling rules the following are considered: sequencing, selection of a piece among those waiting to receive service from a machine and routing decisions concerning the next required workstation. The first two rules (sequencing and job selection) set priorities for jobs waiting in a queue (loading station buffer or workstations input buffers), while the third rule (routing) involves a decision between different routing plans (when there are alternatives). Two FLS have been used for sequencing and priority setting, while a fuzzy MADM technique has been used for the routing problem. The present approach has the advantage of taking into account the multiple objective nature of the FMS control problem through a PI obtained according to fuzzy MADM techniques.

2. WORKING ASSUMPTIONS

The FMS has been modeled according to the following assumptions. Tool management is not considered. Failure of workstations and/or transport systems is not considered. Orders arrive to the FMS as Poisson processes with a fixed inter-arrival time. Production of orders occurs in batches, and the movement of the whole batch is considered, so that batch dimensions are not important. Setup times are independent of the order in which operations are executed, i.e., they are constant and embodied in the operation times of each job (batch). There are as many pallets and fixtures as are needed. The routing of every order is random and directly defined for a workstation, not for the operation, therefore every operation corresponds directly to the workstation that will execute it. There can be multiple routing choices, i.e., at a certain point a job can be equivalently sent to different workstations (as specified in its routing

plan). Loading, unloading and processing times are random. Due dates are assigned according to the total processing time of a job plus some random perturbations. Each workstation can work only one job at a time. The transport system is comprised of automated guided vehicles (AGV), and an AGV can transport only one job at a time. Neither the weight of a piece nor the dimension of a batch affects the speed of AGV, which is assumed to be constant. Every workstation has one input buffer and no output buffer, therefore it will be idle as soon as there is one free AGV that can transport the processed job to another workstation. Delays in accessing the state information are neglected.

3. FUZZY SCHEDULING

The "priority setting" problems (sequencing and piece selection) have been approached using two FLS. Both of these fuzzy logic systems are characterized by: singleton fuzzification, max-product inference (product t-norm and max t-conorm), max rule composition, two antecedents for each rule, one consequent, three triangular membership functions for every antecedent and consequent and modified height defuzzification. Both FLS assign the priority to jobs in queues. The job with the highest priority is the one that will be selected from the queue and, therefore, will either enter the system through the load station (sequencing) or begin to be processed by a workstation (job selection). A complete set of rules have been used. The antecedents used in the sequencing FLS are total processing time (TPT) and due date minus actual time (DD-TN). The antecedents used in the workstation job selection FLS are: processing time in the corresponding workstation (PT) and slack (SL). Each FLS is completely defined once its rules and membership functions are. The consequent membership functions are isosceles triangles defined by three parameters. The antecedent membership functions are defined by seven parameters, therefore, we need seven parameters for each antecedent and nine rules to define each FLS, that is, each FLS is characterized by 23 parameters.

The routing problem has been approached using a fuzzy MADM technique, that is, Saaty [4] criterion as modified by Yager [8], [9]. When a piece has been processed by a particular machine and an AGV is available, the routing controller has to decide which is the next workstation the piece has to go to. The next workstation is chosen by considering three different objectives: low workstation workload, low processing time of the piece for that workstation and low distance of the workstation from the actual piece position. The decision will be made by weighing the degree of satisfaction of each objective. This means that we consider each feasible alternative and take fuzzy measures of the degree of satisfaction of each one of the three objectives. We weigh these measures according to the importance of each criterion, obtaining a "score" for the given alternative. This score represent the overall objective degree of satisfaction and is given by (1):

$$\mu_o(x_j) = [\mu_{C1}(x_{j1})]^{\alpha_1} \cdot [\mu_{C2}(x_{j2})]^{\alpha_2} \cdot [\mu_{C3}(x_{j3})]^{\alpha_3} \quad (1)$$

where $\mu_o(x_j)$ is the overall objective degree of satisfaction corresponding to the j-th alternative, $\mu_{Ci}(x_{ji})$ is the degree of satisfaction of the i-th objective (relatively to the j-th alternative) and α_i its weight. The alternative corresponding to the highest overall objective degree of satisfaction is chosen. The importance of each criterion is given by the weights obtained from a pairwise comparison matrix through the λ_{max} technique. The pairwise comparison matrix (in this case 3×3) usually contains human expert linguistic estimates of pairwise comparisons between the objectives. This decision structure is completely defined once the pairwise comparison matrix and the membership functions for each objective are given. Figure 1 shows the membership function for the low workload and for the low processing time, it is completely defined (for every objective) by four parameters (β, γ, h and θ, the "time constant" of the exponential part of the function). The membership function for low distance is a discrete one and it is arbitrarily assigned. It will be assumed that the pairwise comparison matrix is already specified, e.g., by experts. We only need to define three parameters (β, γ and θ) for each one of the other two objectives.

4. THE REINFORCEMENT LEARNING APPROACH

In the previous section the architecture of a fuzzy scheduler (FS) was presented. There are several parameters that have to be defined in order to have a particular FS, and generally the parameters of a FS suited to a particular FMS would not perform well for another FMS with different characteristics.

Dadone et al. [6] present a simulator-based reinforcement learning approach where a reinforcement learning mechanism adjusts the adaptive controller parameters by evaluating a performance index (PI). The PI provides an evaluation of the performance of the controller in a certain simulation run (in the case of a

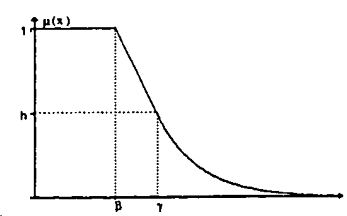

FIGURE 1 - OBJECTIVES SATISFACTION MEMBERSHIP FUNCTION

simulation, i.e. off-line adaptation) or in a particular sliding window (in the case of on-line adaptation). In this particular case we will just consider the design problem (i.e. off-line adaptation); therefore, we consider the plant to be a simulation of the system. The adaptive controller is the FS described above. In our approach the PI evaluation block is characterized by a fuzzy MADM technique ([8], [9]). Following what in [7], [5] and [10], we considered as objectives:

O.1: Low ratio of time in system (TIS) over total processing time (TPT), it should be close to one, that is, the job stays in the system just the minimum time required for it to be processed.

O.2: Low ratio of work in progress (WIP) over the number of workstations (W), it should be close to one; that is, the system does not have excessive inventory costs.

O.3: High resource utilization (MU), it should be close to one, that is, the machines are highly utilized.

O.4: Low tardiness (T), it should be close to zero, that is, orders are not overdue.

After one simulation run, average measures of TIS/TPT, WIP/W, MU and T are fed into the PI evaluation block. Here these measures are passed to the corresponding membership functions to provide the degree of satisfaction of each objective. The different degrees of satisfaction then form the overall control objective degree of satisfaction (i.e., the PI) according to the weighing mechanism described previously for the fuzzy MADM of Section 3. The absolute weights of each objective can be extracted from the corresponding pairwise comparison matrix with the λ_{max} technique. The reinforcement learning mechanism (Figure 3) should be capable of detecting performance improvements and to reward the changes that characterize these improvements. In our case an evolutionary algorithm ([2]), has been used. Every individual is made of 52 components. We will consider each generation made of k individuals. The first generation is obtained from perturbations on an initial FS that we determined "heuristically" through simulations of the system. After the first generation is determined, we run a simulation for each individual and thereby obtain a PI corresponding to every individual. The $k/2$ best will form the next generation together with other $k/2$ individuals determined from them by random perturbations. New generations are formed as off-springs of the best individuals of the previous generation. The algorithm is stopped after a fixed number of generations.

5. RESULTS

To test the described fuzzy scheduler and design procedure, a particular FMS with six machines and three AGV has been considered. The evolutionary algorithm is such that each generation is comprised of ten individuals, and the PI is evaluated from a single simulation of the system made with the same starting seed for every individual and for every generation. The PI is determined according to some selected membership functions, which were determined from simulations of the system with several heuristic scheduling rules by choosing function parameters such that the average of the average measures corresponds to an objective degree of satisfaction of around 0.5-0.6 for the heuristic rules. This was done in order to make it possible to leave some possibilities of improvement for the evolutionary fuzzy scheduler (EFS). The degrees of satisfaction of the four objectives are then weighted according to the weights obtained with the λ_{max} technique from the following pairwise comparison matrix:

$$A = \begin{bmatrix} 1 & 4 & 2 & 1/5 \\ 1/4 & 1 & 1/3 & 1/7 \\ 1/2 & 3 & 1 & 1/5 \\ 5 & 7 & 5 & 1 \end{bmatrix} \qquad (2)$$

In this case $\lambda_{max} = 4.1478$, which is not so far from 4 (therefore the consistency property is not violated for A), and the weights are given by the corresponding "normalized" (i.e., the sum of the components is the order of A, that is, 4) eigenvector w = [0.7646, 0.2273, 0.4976, 2.5105]T. For the fuzzy MADM used in the routing problem the weights for the three objectives, which were obtained in a similar manner, are: **w** = [1.911, 0.7749, 0.3142 $]^T$.

To test the FS and the EFS some heuristic rules commonly used in practice ([10]) have been considered. These are: EDD (earliest due date), FIFO (first-in first-out), LTPT (longest total processing time) and STPT (shortest total processing time) for sequencing; EDD, FIFO, LIFO (last-in first-out), LS (least slack), SPT (shortest processing time) and SPT/TPT (shortest processing time to total processing time ratio) for job selection; and SQL (shortest queue length) and SQW (shortest queue workload) for routing. These rules generally perform well the scheduling task for an FMS, but we cannot say generally which are the best as this depends on the particular FMS. In this case they have been tested through simulations and the best five of all the possible combinations are: E1=(EDD, EDD, SWQ); E2=(FIFO, EDD, SWQ); E3=(STPT, EDD, SWQ); E4=(FIFO, LS, SWQ); E5=(STPT, LS, SWQ).

Running the evolutionary optimization for 400 generations leads to the convergence history shown in Figure 2 where the PI of the best and worse individuals among the five best of each generation is shown. Considering the final generation of the evolutionary algorithm, we take the best five individuals and each of them will correspond to an EFS. Afterwards the heuristic rules, the FS and the EFS have been compared in terms

of their average performances over 100 independent simulations.

Comparisons will be presented in detail; but, generally, we see that the FS has a PI that is around 20 % better than that of the heuristic rules. Through the analysis of performances on the different objectives it is seen that the FS is slightly better than the heuristic rules with respect to TIS and WIP; a bit worse than the heuristic rules with respect to resources utilization, and is better by about 10% than the heuristic rules with respect to tardiness. Both the FS and the EFS are mostly effective in decreasing the tardiness of jobs; this makes sense if we look back at the weights for the objectives. Indeed low tardiness is the most important objective (i.e., has the highest weight), and is therefore the one for which the performance is most optimized. We can also observe how the heuristic rules and the fuzzy scheduling in this case act following two different strategies. Specifically higher values of utilization and tardiness for the heuristic rules suggest the result that more pieces are placed into the system in order to utilize the resources as much as possible, but with the risk of congestion problems. On the other hand, fuzzy scheduling sends fewer pieces into the system so that the system is less utilized, but with no congestion problems. This is more evident if we decrease the inter-arrival time of the orders. If we consider that the optimization was not run for this inter-arrival time, we can conclude that, although there is an obvious performance degradation in the case of lower inter-arrival time, the fuzzy logic based scheduling reacts much better than the heuristic rules without having any explicit term that accounts for that. This suggests that the fuzzy logic scheduler is more robust to congestion problems.

This initial study shows that fuzzy logic can be successfully applied to the scheduling problem of an FMS, but also points to the need for further research.

6. CONCLUSIONS

A fuzzy scheduler for a flexible manufacturing system has been described. Moreover, a design procedure utilizing evolutionary algorithms along with simulations of the plant has also been described. The corresponding evolutionary fuzzy scheduler has been compared to commonly used heuristic rules. The EFS performed better than the heuristic rules according to a multiobjective performance index. Changing the operating conditions (order inter-arrival time) creates some performance degradation, but shows that the FS is significantly better than the heuristic rules.

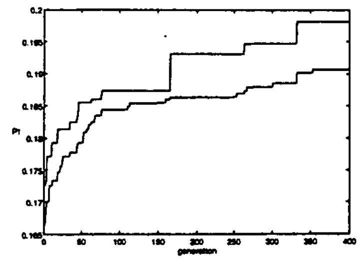

FIGURE 2 - CONVERGENCE HISTORY FOR THE EVOLUTIONARY ALGORITHM

The results in this paper illustrate that the use of fuzzy logic in "intelligent" control of manufacturing systems outperforms standard control techniques. Moreover this paper is one of the few to present a fuzzy scheduler design that satisfies multiple objectives criteria, as typical of the nature of an FMS. Some objectives and their corresponding importance have been considered, but the technique used here to extract the PI and "optimize" the controller can be easily adapted to any plant requirement.

References.

[1] M.P. Fanti, B. Maione, G. Piscitelli and B. Turchiano, "System approach to design generic software for real-time control of flexible manufacturing systems," *IEEE Transactions on Systems, Man and Cybernetics - Part A: Systems and Humans*, 26(2), 1996, 190-202.

[2] D.B. Fogel, *Evolutionary Computation* (New York: IEEE Press, 1995).

[3] J.M. Mendel, "Fuzzy logic systems for engineering: a tutorial," *Proceedings of the IEEE*, 83(3), 1995, 345-377.

[4] T.L. Saaty, "Exploring the interface between hierarchies, multiple objectives and fuzzy sets," *Fuzzy Sets and Systems*, 1, 1978, 57-68.

[5] H. Tempelmeier and H. Kuhn, *Flexible Manufacturing Systems* (John Wiley and sons, 1993).

[6] P. Dadone, H.F. VanLandingham and B. Maione, "Modeling and Control of Discrete Event Dynamic Systems: a Simulator-Based Reinforcement-Learning Paradigm," to appear on *International Journal of Intelligent Control and Systems*, 1997.

[7] N. Viswanadham and Y. Narahari, *Performance modeling of Automated Manufacturing Systems* (Prentice Hall, 1992).

[8] R.R. Yager, "Fuzzy decision making including unequal objectives," *Fuzzy Sets and Systems*, 1, 1978, 87-95.

[9] R.R. Yager, "Multiple objective decision-making using fuzzy sets," *International Journal of Man-Machine Studies*, 9, 1977, 375-382.

[10] H. Young-On, "FMS performance versus WIP under different scheduling rules," *Master's Thesis VPI & SU*, 1994.